Soils and Foundations 2014;54(3):515–522

The Japanese Geotechnical Society

Soils and Foundations

www.sciencedirect.com
journal homepage: www.elsevier.com/locate/sandf

Load–settlement modeling of axially loaded steel driven piles using CPT-based recurrent neural networks

Mohamed A. Shahin*

Department of Civil Engineering, Curtin University, Perth, WA 6845, Australia

Received 29 March 2013; received in revised form 28 November 2013; accepted 25 January 2014
Available online 9 May 2014

Abstract

The design of pile foundations requires good estimations of the pile load-carrying capacity and the settlement. Designs for bearing capacity and settlement have been traditionally carried out separately. However, soil resistance and settlement are influenced by each other, and thus, the design of pile foundations should consider the bearing capacity and the settlement together. This requires that the full load–settlement response of the piles be accurately predicted. However, it is well known that the actual load–settlement response of pile foundations can only be obtained through load tests carried out in-situ, which are expensive and time-consuming. In this technical note, recurrent neural networks (RNNs) were used to develop a prediction model that can resemble the load–settlement response of steel driven piles subjected to axial loading. The developed RNN model was calibrated and validated using several in-situ full-scale pile load tests, as well as cone penetration test (CPT) data. The results indicate that the developed RNN model has the ability to reliably predict the load–settlement response of axially loaded steel driven piles, and thus, can be used by geotechnical engineers for routine design practice.

Keywords: Pile foundations; Load–settlement; Modeling; Recurrent neural networks

1. Introduction

Bearing capacity and settlement are the two main criteria that govern the design process of pile foundations so that safety and serviceability requirements can be achieved. Designs for bearing capacity are carried out by determining the allowable pile load (the service load), which is obtained by dividing the ultimate pile load by an assumed factor of safety. Designs for settlement, on the other hand, consist of obtaining the amount of settlement that occurs when the allowable (service) load on the piles puts stress on the soil, causing the soil to consolidate or compress. In the absence of load–settlement curves, the designs for bearing capacity and settlement have traditionally been carried out separately. However, Fellenius (1988) stated that: "*The allowable load on the pile should be governed by a combined approach considering soil resistance and settlement inseparably acting together and each influencing the value of the other*". In fact, the methods for determining the ultimate (failure) load based on the load–settlement response are the most reliable, provided that the load–settlement curves have been well predicted and simulated. However, there is a strong argument regarding the definition of the ultimate pile load, and multiple criteria have been proposed in literature to interpret the pile load capacity from the pile load–settlement curves. Some methods

*Tel.: +61 8 9266 1822; fax: +61 8 9266 2681.
E-mail address: m.shahin@curtin.edu.au
Peer review under responsibility of The Japanese Geotechnical Society.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.sandf.2014.04.015

result in interpreted ultimate loads that greatly depend on judgement and the shape of the load–settlement curves. For example, the failure load may be defined based on the settlement such as that which causes a settlement equal to 10% of the pile diameter. This load is divided by a nominal factor of safety of 2 to obtain the working load which is used to calculate the settlement. However, if this criterion is applied to certain piles under certain soil conditions (e.g., piles of large diameter in clayey soils), then the settlement at the calculated working load may be excessive (Murthy, 2003). Another criterion, based on the shape of the load–settlement curves, is to plot both the load and the settlement on logarithmic scales. This often leads to two segments of straight lines, and the ultimate load is defined by the point of the maximum curvature. Methods for determining the failure load using the load–settlement curves allow the designer to decide which ultimate load definition should be used, depending on the conditions of the pile and the soil, thus complying with the serviceability requirements.

Good predictions of the load–settlement response of pile foundations need a thorough understanding of the load transfer along the pile length, which is complex, indeterminate, and difficult to quantify (Reese et al., 2006). The actual load–settlement response of pile foundations can only be obtained by carrying out full-scale in-situ load tests, which provide the most precise assessment of the ultimate load capacity. However, full-scale load tests are expensive and time-consuming. Alternatively, the load–settlement response of pile foundations can be estimated using several methods available in literature. However, due to many complexities, these available methods, by necessity, simplify the problem by incorporating several assumptions associated with the factors that affect pile behavior. Therefore, most of the existing methods fail to achieve consistent success in relation to predictions of the pile capacity and the corresponding settlement. In this respect, artificial intelligence techniques, such as artificial neural networks (ANNs), are more efficient as they can resemble the in-situ full-scale pile load tests without the need for any assumptions or simplifications.

ANNs are a data mining statistical approach that has proved its potential in many applications in geotechnical engineering; interested readers are referred to Shahin et al. (2001), where the pre-2001 applications are reviewed in some detail, and Shahin et al. (2009) and Shahin (2013), where the post-2001 applications are briefly examined or acknowledged. In recent years, ANNs have been used with varying degrees of success to predict the axial and lateral bearing capacities of pile foundations in compression and uplift, including driven piles (e.g., Chan et al., 1995; Goh, 1996; Lee and Lee, 1996; Teh et al., 1997; Abu-Kiefa, 1998; Goh et al., 2005; Das and Basudhar, 2006; Pal, 2006; Shahin and Jaksa, 2006; Ahmad et al., 2007; Ardalan et al., 2009; Shahin, 2010; Alkroosh and Nikraz, 2011; Tarawneh, 2013) and drilled shafts (e.g., Goh et al., 2005; Shahin, 2010; Alkroosh and Nikraz, 2011). However, to the author's best knowledge, ANNs have not been previously used for modeling the load–settlement response of pile foundations; the present study will fill in part of this gap.

In this technical note, the feasibility of using one of the ANN techniques, i.e., recurrent neural networks (RNNs), is investigated for modeling the load–settlement response of steel driven piles subjected to axial loading. As mentioned by Briaud et al. (1986), the problem of piles all in sand or all in clay seems to be handled reasonably well by many methods. However, the difficulty arises when the piles are driven through layered soils, especially those with the tip in sand. In the current work, the RNN model is developed for any soil type, including layered soils, and the model works for piles subjected to either compression or uplift loading. To facilitate the use of the developed RNN model for routine use by practitioners, it was translated into an executable program that is made available for interested readers upon request.

2. Overview of recurrent neural networks

The artificial neural networks used in this study are multilayer perceptrons (MLPs) that are trained with the back-propagation algorithm (Rumelhart et al., 1986). A comprehensive description of back-propagation MLPs is beyond the scope of this technical note, but can be found in Fausett (1994). The typical MLP consists of a number of processing elements or nodes that are arranged in layers: an input layer, an output layer, and one or more intermediate layers called hidden layers. Each processing element in a specific layer is linked to the processing element of the other layers via weighted connections. The input from each processing element in the previous layer is multiplied by an adjustable connection weight. The weighted inputs are summed at each processing element, and a threshold value (or bias) is either added or subtracted. The combined input is then passed through a nonlinear transfer function (e.g., sigmoidal or tanh function) to produce the output of the processing element. The output of one processing element provides the input to the processing elements in the next layer. The propagation of information in MLPs starts at the input layer, where the network is presented with a pattern of measured input data and the corresponding measured outputs. The outputs of the network are compared with the measured outputs, and an error is calculated. This error is used with a learning rule to adjust the connection weights so as to minimize the prediction error. The above procedure is repeated with the presentation of the new input and output data until a certain stopping criterion is met. Using the above procedure, the network can obtain a set of weights that produces input–output mapping with the smallest possible error. This process is called "training" or "learning". Once it has been successfully accomplished, the performance of the trained model has to be verified using an independent validation set.

In simulations of the typical nonlinear response of pile load–settlement curves, the current state of the load and the settlement governs the next state of the load and the settlement; thus, recurrent neural networks (RNNs) are recommended. The RNNs proposed by Jordan (1986) imply an extension of the MLPs with current-state units, which are processing elements that remember past activity (i.e., memory units). RNNs then have

two sets of input neurons: plan units and current-state units (see Fig. 1). At the beginning of the training process, the first pattern of input data is presented to the plan units, while the current-state units are set to zero. As mentioned earlier, the training proceeds, and the first output pattern of the network is produced. This output is copied back to the current-state units for the next input pattern of data. RNN model development for the load–settlement response of steel driven piles is described in detail below.

3. Development of RNN model

In this work, the RNN model was developed with the computer-based software package Neuroshell 2, Release 4.2 (Ward, 2007). The data used to calibrate and validate the model were obtained from the literature and included a series of 23 in-situ full-scale pile load–settlement tests reported by Eslami (1996). The tests were conducted at sites with different soil types and geotechnical conditions, ranging from cohesive clays to cohesionless sands including layered soils. The pile load tests included compression and uplift loading conducted on steel driven piles of different shapes (i.e., circular piles with closed toes and *H*-piles with open toes). The piles ranged in diameter from 273 mm to 660 mm with embedment lengths between 9.2 m and 34.3 m. Consequently, the RNN model was developed and validated using data that span the ranges of conditions found in the majority of practical problems.

3.1. Model inputs and outputs

Six factors affecting the capacity of driven piles were presented to the plan units of the RNN as potential model input variables (Fig. 1). These six factors were chosen because they are included in several traditional methods as the most significant factors affecting the load–settlement response of driven pile foundations. These factors represent the pile geometry and the soil properties. The pile geometry includes the pile diameter, D (the equivalent diameter is rather used in the case of *H*-piles as: pile perimeter/π), and the pile embedment length, L. On the other hand, due to the complex geological processes, soils are usually layered (or stratified) and the variation in soil properties must be taken into account by averaging over a sufficient influence zone. Bowles (1997) suggested a number of averaging methods for handling layered soils; a useful averaging technique is the weighted average values adopted in the current study. Consequently, the soil properties considered here are the weighted average cone point resistance over the pile tip failure zone, $\overline{q}_{c-tip}$, the weighted average friction ratio over the pile tip failure zone, $\overline{f}_{R-tip}$, the weighted average cone point resistance over the pile embedment length, $\overline{q}_{c-shaft}$, and the weighted average friction ratio over the pile embedment length, $\overline{f}_{R-shaft}$. The friction ratio, f_R, is the ratio of the cone point resistance, q_c, to the cone sleeve friction, f_s, i.e., $f_R = f_s/q_c$. It should be noted that the weighted averaging method has the advantage of taking into account the impact of different layering thicknesses; this provides a better representation of the variation in of soil properties. It should also be noted that for a single particular soil property, different layering scenarios may lead to the same weighted average of that soil property. For example, a scenario of different layers of cohesive soils may lead to the same weighted average cone resistance, $\overline{q}_c$, to that of another scenario of different layers of cohesionless soils. However, the weighted average cone sleeve friction or friction ratio, $\overline{f}_R$, is unlikely to be the same for both scenarios as the types of soils forming them are different. Consequently, pile capacities are expected to be different depending on the types of soils forming each layering scenario. This agrees well with what one would expect based on the physical meaning.

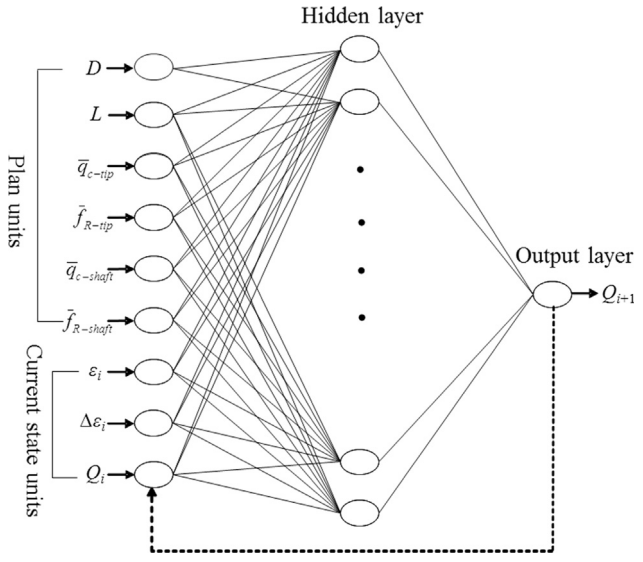

Fig. 1. Architecture of the developed RNN model.

There are some other factors, such as the pile installation type, the load test method, whether the pile tip is open or closed, and the depth of the water table, that contribute to a lesser degree of significance, and thus, are considered secondary and can be neglected (Nejad et al., 2009). The depth of the water table was not considered in this study as the CPT data used here are based on the total resistance (stress). Thus, the effect of the water table has already been accounted for in the measured CPT results. The current state units of the RNN were represented by three input variables, including the normalized axial settlement, $\varepsilon_{a,i}$ (=pile settlement/pile diameter), the increment in axial settlement, $\Delta\varepsilon_{a,i}$, and the pile load, Q_i. The single model output variable is the pile load at the next state of loading, Q_{i+1}.

In this study, an increment in axial settlement that increases by 0.05% was used, in which $\Delta\varepsilon_{a,i}$= (0.1, 0.15, 0.2, …, 1.0, 1.05, 1.1, …) were utilized. As recommended by Penumadu and Zhao (1999), using varying strain increment values results in good modeling capability without the need for a large amount of training data. As the data points needed for the RNN model development were not recorded at the above settlement increments in the original pile load–settlement tests, the load–settlement curves were digitized to obtain the required data points. This was carried out using the computer software Microcal Origin Version 6.0 (Microcal, 1999) and implementing linear interpolation. A range between 14 and 28 training patterns was used to represent a single pile load–settlement test, depending on the maximum incremental settlement values available for each test. It should be noted that the

following conditions were applied to the input and output variables used in the RNN model:

- The pile tip failure zone, over which $\overline{q}_{c-tip}$ and $\overline{f}_{R-tip}$ were calculated, is taken in accordance with Eslami (1996), in which the influence zone extends to 4D below and 8D above the pile toe when the pile toe is located in nonhomogeneous soil of dense strata with a weak layer above it (see Fig. 2(a)). Also, in nonhomogeneous soil, when the pile toe is located in weak strata with a dense layer above it, the influence zone extends to 4D below and 2D above the pile toe (Fig. 2(b)). In homogeneous soil, however, the influence zone extends to 4D below and 4D above the pile toe (Fig. 2(c)).
- The values for both the cone point resistance and the friction ratio were incorporated as model inputs, allowing the soil type (classification) to be implicitly considered in the RNN model.
- Several CPT tests used in this work include mechanical, rather than electric, CPT data. Thus, it was necessary to convert the mechanical CPT readings into equivalent electric CPT values as the electric CPT is the one that is commonly used at present. This is carried out for the cone point resistance using the following correlation proposed by Kulhawy and Mayne (1990):

$$\left(\frac{q_c}{p_a}\right)_{Electric} = 0.47\left(\frac{q_c}{p_a}\right)^{1.19}_{Mechanical} \quad (1)$$

- For the cone sleeve friction, the mechanical cone gives a higher reading than the electric cone in all soils with a ratio in sands of about 2, and 2.5–3.5 for clays (Kulhawy and Mayne, 1990). In the current work, a ratio of 2 was used for sands and 3 for clays.

3.2. Data division and pre-processing

The next step in the development of the RNN model is to divide the available data into subsets. In this work, the data

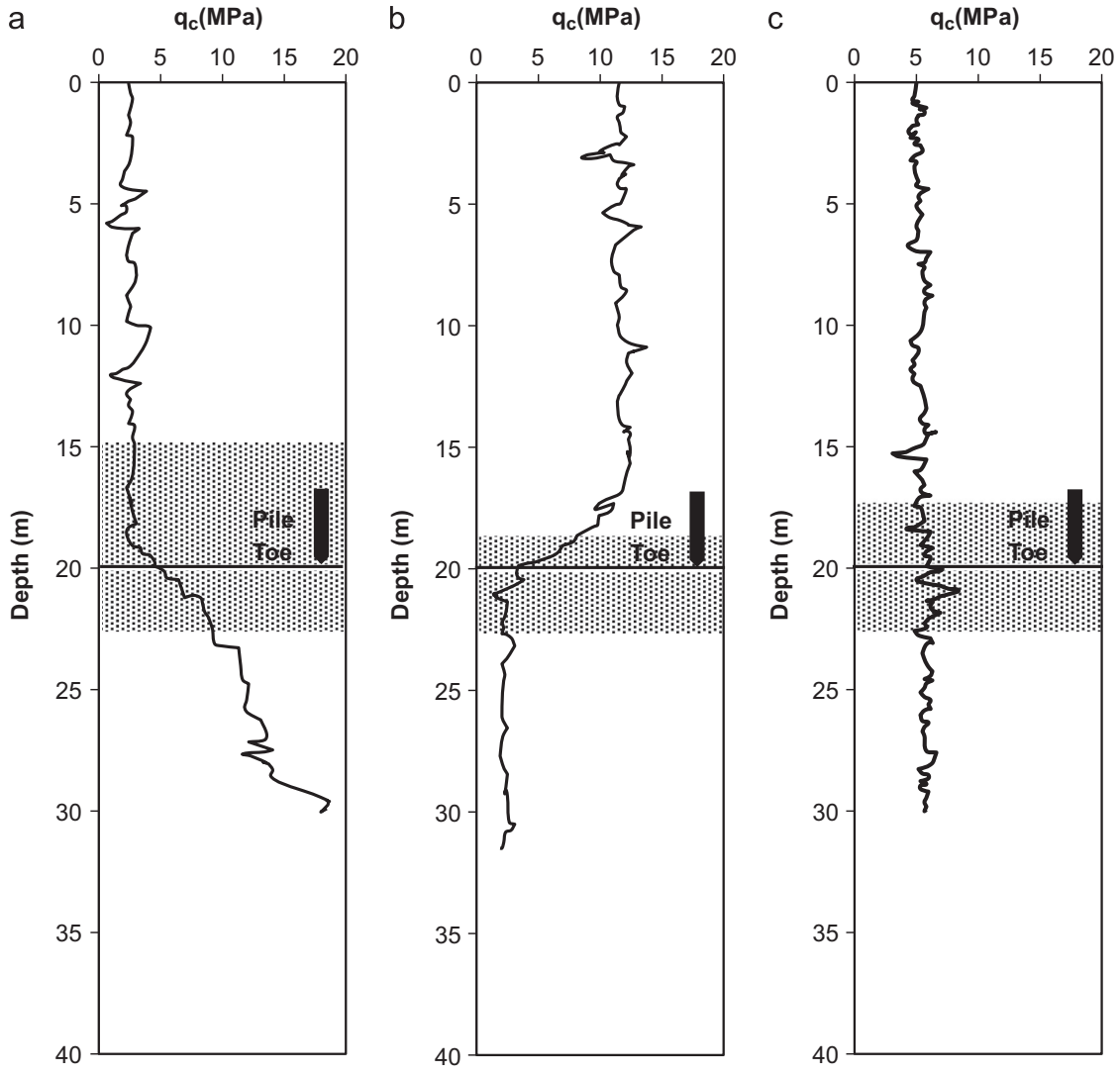

Fig. 2. Pile tip failure zones: (a) nonhomogeneous soil 8D/4D; (b) nonhomogeneous soil 2D/4D; and (c) homogeneous soil 4D/4D.

were randomly divided into two sets: a training set for model calibration and an independent validation set for model verification. As recommended by Masters (1993) and Shahin et al. (2004), the data were divided in such a way that the training and the validation sets would be statically consistent, and thus, would represent the same statistical population. In total, 20 in-situ pile load tests were used for model training and three tests were used for model validation. A summary of the data used in the training and the validation sets, as well as the minimum values, the maximum values, the ranges, and the averages, is given in Table 1. Once the available data were divided into subsets, the input and the output variables were pre-processed. In this step, the variables were scaled between 0.0 and 1.0 to eliminate their dimensions and to ensure that all variables would receive equal attention during training.

3.3. Network architecture and optimization of internal parameters

Following the data division and the pre-processing, the optimum model architecture (i.e., the number of hidden layers and the corresponding number of hidden nodes) must be determined. It should be noted that, as investigated by Hornik et al. (1989) and Cybenko (1989), a network with one hidden layer is sufficient for approximating any continuous function provided that adequate connection weights are used. Hecht-Nielsen (1989) also provided proof that a single hidden layer of neurons is sufficient to model any solution surface of practical interest. Consequently, only one hidden layer was used in the current study. The optimal number of hidden nodes was obtained by a trial-and-error approach in which the network was trained with a set of random initial weights and a fixed learning rate of 0.1, a momentum term of 0.1, a tanh transfer function in the hidden layer nodes, and a sigmoidal transfer function in the output layer nodes. The following numbers of hidden layer nodes were then utilized: 2, 3, 4, …, and $(2I+1)$, where I is the number of input variables. It should be noted that $(2I+1)$ is the upper limit for the number of hidden layer nodes needed to map any continuous function for a network with I inputs, as discussed by Caudill (1988). To obtain the optimum numbers of hidden layer nodes, it is important to strike a balance between having sufficient free parameters (connection weights) to enable representation of the function to be approximated and not having too many, so as to avoid overtraining (Shahin and Indraratna, 2006).

The criterion used to terminate the training process was as follows. The errors between the actual and the predicted values for all outputs in the training set over all patterns were monitored until no significant improvement in the errors occurred. This was achieved for approximately 10,000 training cycles (epochs). Once training had been accomplished, the errors between the actual and the predicted outputs in the validation set were determined for all trained models so that the optimal model could be selected. The model that performed the best in both the training and the validation sets was considered to be optimal. Fig. 3(a) and (b) shows the impact of the number of hidden layer nodes on the root mean squared error (*RMSE*) and the mean absolute error (*MAE*), respectively, of the trained models. The *RMSE* and the *MAE* will be explained in the next section. It can be seen that the network with 10 hidden layer nodes has the lowest prediction error in both the training and the validation sets, and thus, can be considered optimal. As a result of training, the optimal network produced 9×10 weights and 10 bias values connecting the input layer to the hidden layer and 10×1 weights and one bias value connecting the hidden layer to the output layer.

Table 1
Summary of the data used for development of the RNN model.

Test no.	D (mm)	L (m)	$\overline{q}_{c-tip}$ (MPa)	$\overline{f}_{R-tip}$ (%)	$\overline{q}_{c-shaft}$ (MPa)	$\overline{f}_{R-shaft}$ (%)	RNN status[a]	Loading type[b]
1	300	16.2	20.0	0.50	17.5	0.46	T	C
2	455	12.0	0.0	0.00	15.8	0.38	T	U
3	455	11.3	0.0	0.00	15.0	0.40	T	U
4	609	34.3	8.0	0.75	5.2	1.15	T	C
5	273	22.5	18.3	1.09	8.7	0.69	T	C
6	660	18.2	10.0	0.60	9.0	0.67	T	C
7	324	31.1	2.5	2.00	5.6	0.45	T	C
8	455	11.3	0.0	0.00	15.0	0.40	T	C
9	300	28.4	1.0	2.00	2.5	1.20	T	C
10	324	13.7	1.0	2.00	2.1	0.71	T	C
11	273	22.5	0.0	0.00	8.7	0.69	T	U
12	350	11.1	0.0	0.00	15.0	0.40	T	U
13	273	13.0	0.0	0.00	2.5	1.60	T	U
14	455	16.8	0.0	0.00	17.7	0.45	T	U
15	400	14.6	0.0	0.00	15.0	0.40	T	U
16	300	31.4	1.0	2.00	2.5	1.20	T	C
17	450	15.2	1.0	2.00	3.3	2.12	T	C
18	273	9.2	6.5	0.31	6.3	0.32	T	C
19	273	15.2	4.9	1.63	5.3	0.75	T	C
20	330	10.0	4.0	1.00	6.0	1.00	T	C
21	300	11.0	0.0	0.00	15.0	0.40	V	U
22	350	14.4	20.0	0.50	16.5	0.48	V	C
23	400	14.6	19.5	0.51	16.5	0.48	V	C
Minimum	273	9.2	0.0	0.00	2.1	0.32	–	–
Maximum	660	34.3	20.0	2.00	17.7	2.12	–	–
Range	387	25.1	20.0	2.00	15.6	1.80	–	–
Average	376	17.9	3.9	0.80	8.9	0.77	–	–

[a]T: training; V: validation.
[b]C: compression; U: uplift.

3.4. Optimal model performance and validation

The performance of the optimal RNN model in the training and the validation sets is given numerically in Table 2. It can be seen that four different standard performance measures were used, including the coefficient of correlation, r, the coefficient of efficiency, E, the root mean squared error, *RMAE*, and the mean absolute error, *MAE*. The formulas for these four measures are as follows:

$$r = \frac{\sum_{i=1}^{N}(O_i-\overline{O})(P_i-\overline{P})}{\sqrt{\sum_{i=1}^{N}(O_i-\overline{O})^2\sum_{i=1}^{N}(P_i-\overline{P})^2}} \quad (2)$$

$$E = 1-\frac{\sum_{i=1}^{N}(O_i-P_i)^2}{\sum_{i=1}^{N}(O_i-\overline{O})^2} \quad (3)$$

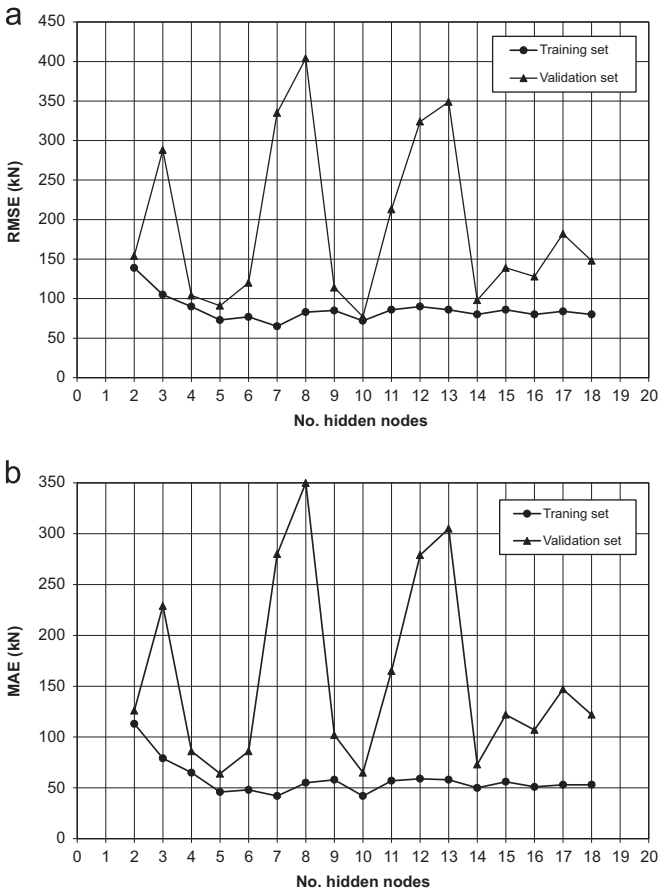

Fig. 3. Effect of number of hidden nodes on the RNN model performance: (a) *RMSE* and (b) *MAE*.

Table 2
Performance results of the optimal RNN model.

Data sets	Performance measures			
	r	*E*	*RMSE* (*kN*)	*MAE* (*kN*)
Training	0.997	0.993	72	42
Validation	0.992	0.973	77	65

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} |O_i - P_i|}{N}} \quad (4)$$

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |O_i - P_i| \quad (5)$$

where N is the number of data points presented in the model, O_i and P_i are the observed and the predicted outputs, respectively, and $\overline{O}$ and $\overline{P}$ are the mean of the predicted and the observed outputs, respectively.

The coefficient of correlation, r, is a measure that is used to determine the relative correlation between the predicted and the observed outputs. However, as indicated by Das and Sivakugan (2010), r sometimes may not necessarily indicate a better model performance due to the tendency of the model to deviate toward higher or lower values, particularly when the data range is very wide and most of the data are distributed about their mean. Consequently, the coefficient of efficiency, E, was used as it can give unbiased estimates and may be a better measure for model performance. The *RMAE* is the most popular error measure; it has the advantage that large errors receive much greater attention than small errors (Hecht-Nielsen, 1990). However, as indicated by Cherkassky et al. (2006), there are situations when the *RMSE* cannot guarantee that the model performance is optimal; thus, the mean absolute error, *MAE*, was also used. The *MAE* eliminates the emphasis given to large errors, and is a desirable measure when the data evaluated are smooth or continuous, which is the case in the current study. The performance measures in Table 2 indicate that the optimum RNN model performs well and has good prediction accuracy in both the training and the validation sets. Table 2 also indicates that the RNN model has as consistent a performance in the validation set as in the training set.

The performance of the optimal RNN model in the training and the validation sets was further investigated graphically, as shown in Figs. 4 and 5, respectively. It should be noted that, for brevity, only five of the most appropriate simulation results in the training set are given in Fig. 4. These five simulations were chosen because they reflect the entire range in in-situ pile load–settlement tests used in this study. As can be seen in Figs. 4 and 5, an excellent agreement between the actual pile load tests and the RNN model predictions is obtained for both the training and the validation sets. The nonlinear relationships of the load–settlement response are well predicted, and the results demonstrate that the RNN model is strongly capable of simulating the behavior of steel driven piles quite well.

3.5. Model robustness and sensitivity analyses

To further examine the generalization ability (or robustness) of the RNN model, sensitivity analyses were carried out to investigate the response of the RNN predicted pile behavior to a set of hypothetical input data that lie in the range of the data used for model training. For example, to investigate the effect

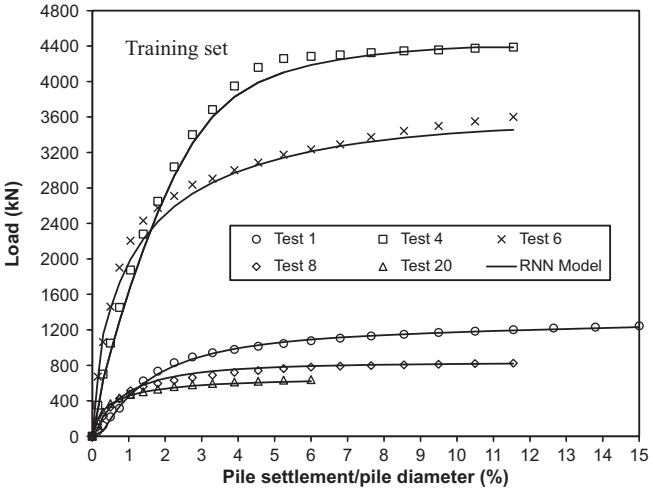

Fig. 4. Some simulation results of the RNN model in the training set.

of one parameter, such as pile diameter D, all other input variables were set to selected constant values, while D was allowed to change. The inputs were then accommodated in the RNN model and the predicted pile load versus settlement response was calculated. This process was repeated for the next input variable and so on, until the model response was examined for all inputs. The robustness of the RNN model was determined by examining how well the predictions compare with the available geotechnical knowledge and experimental data.

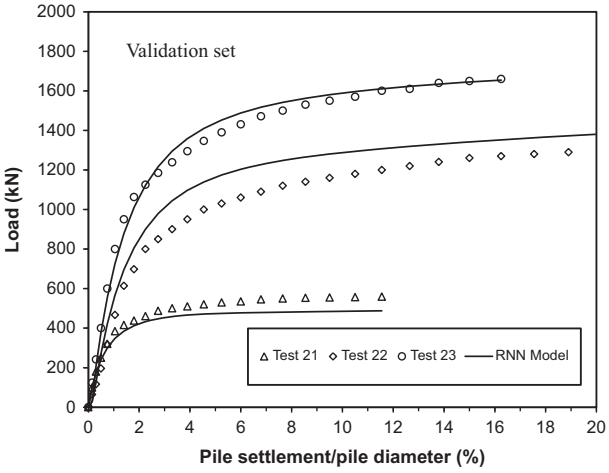

Fig. 5. Simulation results of the RNN model in the validation set.

The results of the sensitivity analyses are shown in Fig. 6; they indicate that the predicted pile behavior by the RNN model is in good agreement with what one would expect based on the underlying physical meaning and with published experimental results. For example, it can be observed that the ultimate pile capacity increases with the increase in pile diameter, pile embedment length, soil resistance at the pile tip, and soil resistance at the pile length. The above results confirm the predictive ability of the developed RNN model in reflecting the role of important factors affecting pile behavior, which indicates that the model is robust and can thus be used with confidence.

4. Conclusion

The work presented in this technical note has used a series of full-scale in-situ pile load–settlement tests and CPT data collected from literature to develop a recurrent neural network (RNN) model for simulating the load–settlement response of steel driven piles. The graphical comparison of the load–settlement curves between the RNN model and the experiments showed an excellent agreement and indicated that the RNN model can capture the highly non-linear load–settlement response of steel driven piles reasonably well. To facilitate the use of the developed RNN model, it was translated into an executable program using the MATLAB code, which is made available for interested readers upon request.

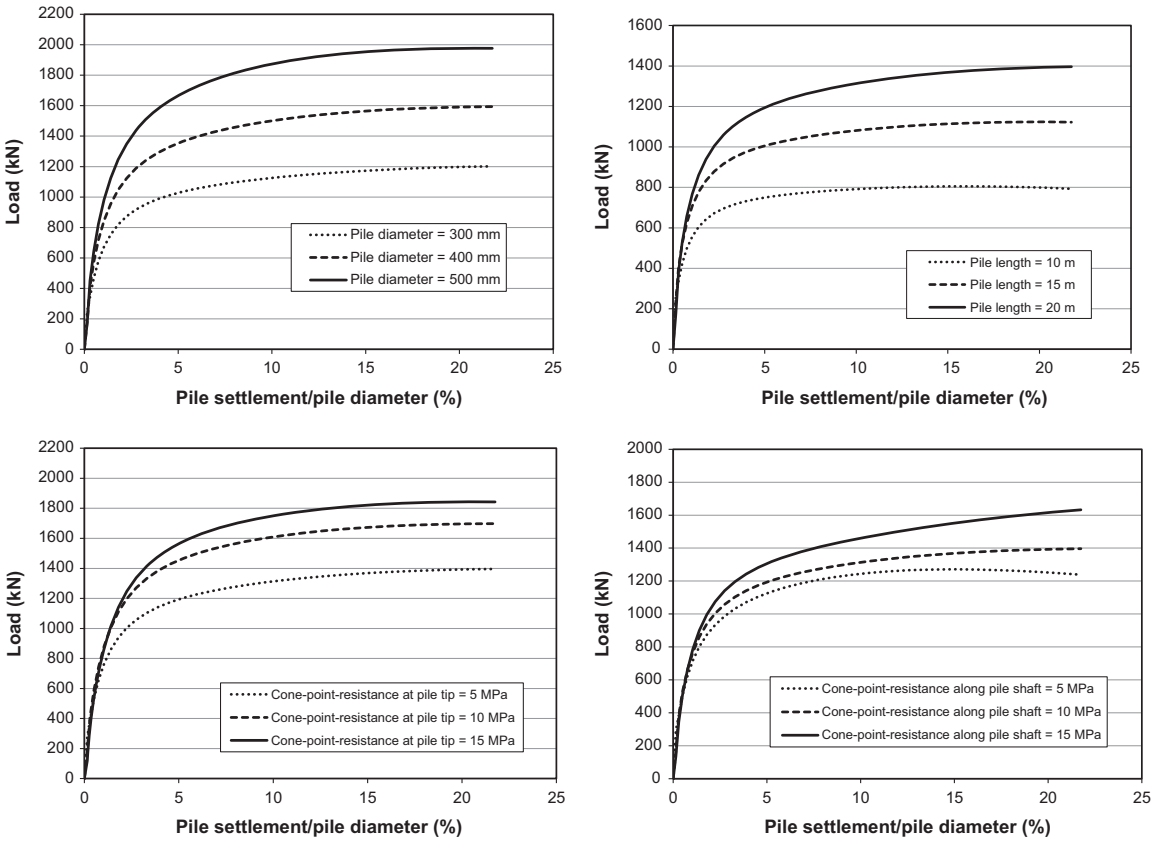

Fig. 6. Sensitivity analyses to test the robustness of the RNN model.

It is worthwhile noting that predictions from ANN models are better when used for ranges in input variables similar to those utilized in model training. This is because ANNs work better in interpolation than extrapolation. Consequently, the developed RNN model performs the best when it is used for the ranges in values of the inputs shown in Table 1. However, the values given in Table 1 span the ranges of conditions found in the majority of practical problems. It should be noted that the developed RNN model, like all other available pile settlement and bearing capacity models, was not developed to deal with very special cases at highly complicated sites which can give surprising results that are hard to explain. However, the model is valid for the common cases of site conditions containing single or multilayer cohesive and/or cohesionless soils. In addition, the model has the advantage that it can always be updated in the future by presenting new training examples of wider ranges, as new data become available. Overall, it is evident from the results of this study that the developed RNN model is robust and can be used with confidence.

References

Abu-Kiefa, M.A., 1998. General regression neural networks for driven piles in cohesionless soils. J. Geotech. Geoenviron. Eng. 124 (12), 1177–1185.

Ahmad, I., El Naggar, H., Kahn, A.N., 2007. Artificial neural network application to estimate kinematic soil pile interaction response parameters. Soil Dyn. Earthq. Eng. 27 (9), 892–905.

Ardalan, H., Eslami, A., Nariman-Zadeh, N., 2009. Piles shaft capacity from CPT and CPTU data by polynomial neural networks and genetic algorithms. Comput. Geotech. 36 (4), 616–625.

Alkroosh, I., Nikraz, H., 2011. Simulating pile load–settlement behavior from CPT data using intelligent computing. Cent. Eur. J. Eng. 1 (3), 295–305.

Briaud, J.L., Tucker, L.M., Anderson, J.S., Perdomo, D., Coyle, H.M., 1986. Development of an improved Pile Design Procedure for Single Piles in Clays and Sands. Research Report 4981-1. Civil Engineering, Texas A&M University, Texas.

Bowles, J.E., 1997. Foundation Analysis and Design. McGraw-Hill, New York.

Caudill, M., 1988. Neural networks primer, Part III. AI Expert 3 (6), 53–59.

Cybenko, G., 1989. Approximation by superpositions of a sigmoidal function. Math. Control Signals Syst. 3, 303–314.

Chan, W.T., Chow, Y.K., Liu, L.F., 1995. Neural network: an alternative to pile driving formulas. Comput. Geotech. 17, 135–156.

Cherkassky, V., Krasnopolsky, D.P., Valdes, J., 2006. Computational intelligence in earth sciences and environmental applications: issues and challenges. Neural Netw. 19, 113–121.

Das, S.K., Basudhar, P.K., 2006. Undrained lateral load capacity of piles in clay using artificial neural network. Comput. Geotech. 33 (8), 454–459.

Das, S.K., Sivakugan, N., 2010. Discussion of: intelligent computing for modeling axial capacity of pile foundations. Can. Geotech. J. 47, 928–930.

Eslami, A., 1996. Bearing Capacity of Piles from Cone Penetration Test Data (Ph.D. thesis). University of Ottawa, Ottawa, Ontario.

Fellenius, B.H., 1988. Unified design of piles and pile groups. Transp. Res. Rec. 1169, 75–81.

Fausett, L.V., 1994. Fundamentals Neural Networks: Architecture, Algorithms, and Applications. Prentice-Hall, Englewood Cliffs, New Jersey.

Goh, A.T.C., 1996. Pile driving records reanalyzed using neural networks. J. Geotech. Eng. 122 (6), 492–495.

Goh, A.T., Kulhawy, F.H., Chua, C.G., 2005. Bayesian neural network analysis of undrained side resistance of drilled shafts. J. Geotech. Geoenviron. Eng. 131 (1), 84–93.

Hecht-Nielsen, R., 1989. Theory of the backpropagation neural network. In: Proceedings of the International Joint Conference on Neural Networks. Washington, DC, pp. 593–605.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Netw. 2, 359–366.

Hecht-Nielsen, R., 1990. Neurocomputing. Addison-Wesley Publishing Company, Reading, MA.

Jordan, M.I., 1986. Attractor dynamics and parallelism in a connectionist sequential machine. In: Proceedings of the 8th Annual Conference of the Cognitive Science Society. Amherst, MA, pp. 531–546.

Kulhawy, F.H., Mayne, P.W., 1990. Manual on Estimating Soil Properties for Foundation Design. Report EL-6800, Electric Power Research Institute, Palo Alto, CA.

Lee, I.M., Lee, J.H., 1996. Prediction of pile bearing capacity using artificial neural networks. Comput. Geotech. 18 (3), 189–200.

Masters, T., 1993. Practical Neural Network Recipes in C++. Academic Press, San Diego, California.

Microcal, 1999. Microcal Origin Version 6.0. Microcal Software, Inc., Northampton, MA.

Murthy, V.N.S., 2003. Geotechnical Engineering: Principles and Practices of Soil Mechanics and Foundation Engineering. Marcel Dekker, Inc., New York.

Nejad, F.P., Jaksa, M.B., Kakhi, M., McCabe, B.A., 2009. Prediction of pile settlement using artificial neural networks based on standard penetration test data. Comput. Geotech. 36 (7), 1125–1133.

Penumadu, D., Zhao, R., 1999. Triaxial compression behavior of sand and gravel using artificial neural networks (ANN). Comput. Geotech. 24 (3), 207–230.

Pal, M., 2006. Support vector machine-based modeling of seismic liquefaction potential. Int. J. Numer. Anal. Methods Geomech. 30 (10), 983–996.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representation by error propagation. In: Rumelhart, D.E., McClelland, J. L. (Eds.), Parallel Distributed Processing. MIT Press, Cambridge.

Reese, L.C., Isenhower, W.M., Wang, S.T., 2006. Analysis and Design of Shallow and Deep Foundations. John Wiley & Sons, New Jersey.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2001. Artificial neural network applications in geotechnical engineering. Aust. Geomech. 36 (1), 49–62.

Shahin, M.A., Maier, H.R., Jaksa, M.B., 2004. Data division for developing neural networks applied to geotechnical engineering. J. Comput. Civ. Eng. 18 (2), 105–114.

Shahin, M.A., Indraratna, B., 2006. Modeling the mechanical behavior of railway ballast using artificial neural networks. Can. Geotech. J. 43 (1), 1144–1152.

Shahin, M.A., Jaksa, M.B., 2006. Pullout capacity of small ground anchors by direct cone penetration test methods and neural networks. Can. Geotech. J. 43 (6), 626–637.

Shahin, M.A., Jaksa, M.B., Maier, H.R., 2009. Recent advances and future challenges for artificial neural systems in geotechnical engineering applications. J. Adv. Artif. Neural Syst. (Article ID 308239, 9 pp). http://dx.doi.org/10.1155/2009/308239.

Shahin, M.A., 2010. Intelligent computing for modeling axial capacity of pile foundations. Can. Geotech. J. 47 (2), 230–243.

Shahin, M.A., 2013. Artificial intelligence in geotechnical engineering: applications, modeling aspects, and future directions. In: Yang, X., Gandomi, A.H., Talatahari, S., Alavi, A.H. (Eds.), Metaheuristics in Water, Geotechnical and Transport Engineering. Elsevier Inc., London, pp. 169–204.

Teh, C.I., Wong, K.S., Goh, A.T.C., Jaritngam, S., 1997. Prediction of pile capacity using neural networks. J. Comput. Civ. Eng. 11 (2), 129–138.

Tarawneh, B., 2013. Pipe pile setup: database and prediction model using artificial neural network. Soils Found. 53 (4), 607–615.

Ward, 2007. NeuroShell 2 Release, 4. Ward Systems Group, Inc. (MA).